Journal Pre-proof

How does the traffic behavior change by using SUMO traffic generation tools

Pablo Barbecho Bautista, Luis Urquiza Aguiar, Mónica Aguilar Igartua

PII: S0140-3664(21)00360-1
DOI: https://doi.org/10.1016/j.comcom.2021.09.023
Reference: COMCOM 6905

To appear in: *Computer Communications*

Received date : 9 April 2021
Revised date : 8 July 2021
Accepted date : 20 September 2021

Please cite this article as: P.B. Bautista, L.U. Aguiar and M.A. Igartua, How does the traffic behavior change by using SUMO traffic generation tools, *Computer Communications* (2021), doi: https://doi.org/10.1016/j.comcom.2021.09.023.

How does the traffic behavior change by using SUMO traffic generation tools☆

Pablo Barbecho Bautista[a,1,*], Luis Urquiza Aguiar[b], Mónica Aguilar Igartua[a]

[a]Dep. of Network Engineering. Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
[b]Dep. of Electronics, Telecommunication and Networks. Escuela Politécnica Nacional (EPN), Quito, Ecuador

Abstract

Simulations are the traditional approach used by the research community to evaluate mobile ad hoc networks. Mainly, vehicular ad hoc networks (VANETs) are a particular type of mobile ad hoc networks that raise specific technical challenges. When assessing VANETs, it is crucial to use realistic mobility models and traffic demand to produce meaningful results. In this context, vehicular traces affect vehicles' signal strengths, radio interference, and channel occupancy. This paper provides a thorough analysis of the influence of using the different SUMO's traffic demand generation tools on mobility and node connectivity. Using the data traffic in the district of Gracia in Barcelona (Spain), we analyze the generated traffic demand in terms of traffic measures: (i) traffic intensity, (ii) trip time/distance, (iii) emissions, and (iv) re-routing capabilities. This last feature allow cars to re-compute their routes in front of congestion situations. Then, we analyze the available tools in terms of resources usage (CPU, RAM, disk). Lastly, we analyze the node's connectivity using well-known graph metrics. Our results provide insights into the behavior of the vehicle's mobility and the nodes' connectivity of SUMO demand generation tools. Additionally, we propose an automatized tool that facilitates researchers the generation of synthetic traffic based on real data.

Keywords: Automatic routing, traffic demand, SUMO, VANET.

1. Introduction

Nowadays, simulation is the main approach that researchers follow to assess the performance of wireless networks. Particularly, while the deployment of real test scenarios is feasible in most cases (e.g., sensor networks), simulation techniques are commonly used to evaluate vehicular networks and new emerging services for intelligent transportation systems (ITS). In this regard, the simulation of vehicular communications usually involves the coupling of a network simulator and a traffic simulator. Network simulators implement the stack of protocols for vehicular communications (e.g., Veins [1] following the IEEE WAVE specification, Artery [2] following the ETSI ITS-G5 specification). Traffic simulators implement the traffic-related elements (e.g., road network, traffic demand, mobility models) (e.g., VISSIM, MATSim, SUMO, TranSims). Here, realistic mobility models allows simulators to interact with obstacles along the route, traffic/weather conditions, and drivers behavior.

Hereabouts, wireless system researchers put their effort into the network modelling without considering the effect of the traffic during the simulation. In this context, jointly considering realistic mobility models and traffic demand generation are keys to obtain accurate results when evaluating vehicular networks.

The main elements in order to generate the traffic simulation are (i) the network data consisting on roads and intersections, a.k.a edges and junctions; (ii) traffic infrastructure containing traffic lights elements and logic; (iii) vehicles' type consisting on a description of vehicle's characteristics (e.g., gas/diesel, passenger/bus); and (iv) vehicles' traces that vehicles will follow during the simulation (e.g., routes, trips, flows). Regarding this last, we can differentiate two main groups: (i) vehicles' traces defined by hand, mostly used when evaluating simple map topologies (e.g., highway, Manhattan); and (ii) vehicles' traces defined using traffic demand generation tools [3], used with large size maps or complex topologies (e.g., city map).

In this regard, a wide-spread road traffic simulator is Simulation of Urban MObility (SUMO) [4]. SUMO is an open-source space-continuous road traffic simulator commonly used for testing vehicular networks and ITS applications.

Most modern vehicles are equipped with real-time navigation services allowing drivers to better use the available road capacity. Besides, drivers can be informed through the vehicular network about the road network's events during their trips. In this context, most drivers are familiarized with the road network and, therefore, can decide to try an alternative path during their trips. This behavior is emulated in SUMO utilizing an additional feature, called *re-routing device*. If vehicles are equipped with a re-routing device, they can change their current routes, taking into account the road network's current state. In this sense, SUMO includes capabilities for re-routing vehicles dynamically. This approach impacts the overall road traffic

☆This work was partly supported by the Spanish Government through project TEC2017-84197-C4-3-R (Secure sMArt Grid using Open Source Intelligence, MAGOS) and SENESCYT.

*Corresponding author

Email addresses: pablo.barbecho@upc.edu (Pablo Barbecho Bautista), luis.urquiza@epn.edu.ec (Luis Urquiza Aguiar), monica.aguilar@upc.edu (Mónica Aguilar Igartua)

[1]Is recipient of a Ph.D. grant from Secretaria Nacional de Educación Superior, Ciencia y Tecnología (SENESCYT), Ecuador.

conditions (i.e., the road congestion level). Consequently, the network connectivity is also affected.

This paper presents a through comparison of SUMO traffic demand generation tools OD2trips, DUArouter, DUAIterate, MArouter, and Randomtrips in terms of the (i) traffic intensity, (ii) trip time/length, and (iii) emissions. Results are compared against real traffic data [5], [6] as well as EURO 6 emissions directives [7] and real-world emission [8]. Also, we present a network evaluation that offers researchers more informed choices when deciding SUMO re-routing configurations. For this, we take advantage of graph theory concepts. Firstly, we perform an offline analysis transforming the vehicular traces into snapshots at different simulation times to latter evaluate the network employing well-known graph metrics. Secondly, we propose an automatized tool that facilitates the generation of synthetic traffic based on real data traffic. Finally, we conduct an extensive performance evaluation to support our analysis and discussion.

The remainder of the paper is organized as follows: Section 2 presents a comparison of related work. Section 3 presents an overview of the simulation tools adopted for the analysis, highlighting the SUMO traffic demand generation tools and re-route functionality. Later, Section 4 shows details of the simulation settings, re-routing capabilities, and the performance evaluation of the different traffic generation tools, including a comparison of resources used by each tool (i.e., CPU, RAM, disk). In Section 5, we investigate the effect of vehicles' re-routing on the nodes' connectivity. Later, Section 6 describes an open-source tool intended to facilitates traffic demand generation. Finally, in Section 7, conclusions and future perspectives are presented.

2. Related work

Over the last few years, the vehicular networking community has proposed many proposals, applications and services. One recurrent problem is to develop a reliable and usable mobility scenario to evaluate services for intelligent transportation systems (ITS). In this context, several works present realistic simulation scenarios derived from real traffic statistics and mobility patterns. Table 1 summarizes some of the salient features of the proposals found in the literature.

Regarding to studies intended to evaluate mobility scenarios (see Table 1), in [9], a realistic vehicular scenario is proposed called *Luxembourg SUMO Traffic* (LuST) scenario (urban mid-size). Codecá. et al. built LuST from the road network of a real city imported from OpenStreetMaps (OSM) [16]. In that work, the authors use traffic statistics of Luxembourg city to calibrate a full-day traffic demand generated using the Activitygen SUMO tool [17]. This last corresponds to an activity-based traffic generation based on the population and considering everyday trips (e.g., bus lines). Then, vehicle routes are generated using SUMO DUARouter tool, see Table 1. The authors evaluate the traffic demand generated considering that 70% of the vehicles' includes re-routing capabilities (i.e., change their routes). In [10], a mixed traffic (autonomous/classic vehicles) analysis is presented. That work is intended to validate SUMO car-following and lane change models for autonomous vehicles. Traffic mobility is generated for an Austrian three-lane highway (Vienna-Klagenfurt direction) based on real-traffic measurements using a single lane routing (i.e., same route for all the vehicles), see Table 1. Then, the effects of increasing the amount of autonomous vehicles (scaling factor in Table 1) are analyzed concluding that a conservative time headway of autonomous vehicles compared to human drivers cause a road capacity reduction.

In [11], the Monaco SUMO Traffic (MoST) scenario includes a multimodal scenario that integrates vulnerable users (e.g., pedestrians, bicycles) intended to study advanced parking management solutions. It is based on synthetic activity-based mobility traces generated for the Principality of Monaco (urban mid-size map). Here, vehicle routes are generated for 10 hr simulation using the DUARouter tool, see Table 1. In [12] a heuristic traffic modelling is proposed intended to evaluate the impact of incorporating additional traffic (scaling factor) on the cities traffic congestion levels. The traffic is modeled based on induction loop measurements (i.e., measurements of the inflow and outflow of the roads) of the most relevant streets/avenues in Valencia, Cologne, and Bologna cities. This information is used later as the input of origin/destination (O/D) matrices to generate vehicles routes using the DFRouter SUMO tool, see Table 1.

Concerning the vehicular network connectivity studies, in [13], the authors propose a Real Vehicular Wireless Network Model (RVWNM) intended for the evaluation of VANETs capacity through graph theory concepts. The proposed model is constructed utilizing a Euclidean planar graph and an interference relationship graph. First, the authors modeled urban area structures (Helsinki city) as grid-based structures using a Euclidean graph, where vertices represent points in the plane, and the Euclidean distance between those points means the edges. The authors propose a modified version of the RandomTrips SUMO tool (see Table 1), where origin points (home-point) of the route are located near to populated places (social-proximity in Table 1). In [14] and [15] authors evaluate the impact of the vehicles' density on the performance of well-known topology-based and road-based vehicular traffic routing (RBVT) protocols for VANETs, respectively. Particularly, RBVT uses real-time vehicular traffic information to create road-based paths with a high probability of connectivity among them to route the packet. Those works consider urban small-size real city maps (Morocco and Los Angeles) and a traffic demand generated using a single SUMO tool (not specified), see Table 1. Although the authors consider real vehicular traffic information, realistic mobility features as online route updates (i.e., re-routing) are not considered, see Table 1. Besides, when evaluating road-based routing protocols, it is paramount to select the appropriate traffic demand generation tool in order to obtain accurate results.

In Table 1, we note that the existing works assume only a single routing tool (e.g., DUARouter, Random Trips). Besides, the impact of re-routing capabilities (i.e., vehicles can change their routes online) and the traffic scaling factor parameters are not always considered. In this work, we present an evaluation of the OD2trips, DUArouter, DUAIterate, MArouter, and Ran-

Table 1: Classification of SUMO tools in the literature. (*) means modified version; STG (SUMO traffic generation tool) is introduced in section Section 6.

Ref.	Study	Traffic statistics	Map	Mobility Features	Routing Tools
LuST [9]	Mobility	Activity-Based	Urban mid-size	Re-routing	DUARouter
[10]	Mobility	Road sensors	Highway	Scaling factor Multimodel transport	Single lane
MoST [11]	Mobility	Activity-Based	Urban mid-size	Multimodal transport	DUARouter
[12]	Mobility	Road sensors	Urban mid-size	Scaling factor	DFRouter
RVWNM [13]	Connectivity	Social-Proximity	Grid-like small-size	None	RandomTrips*
[14], RBVT [15]	Connectivity	-	Urban small-size	None	-
STG	Mobility Connectivity	Road sensors	Urban mid-size	Re-routing Scaling factor	OD2Trips, DUARouter, MARouter, RandomTrips, DUAIterate

domtrips SUMO tools to provide researchers a clear view of the impact on the traffic mobility of each tool (see STG in Table 1). Also, we claim that by allowing vehicles' re-routing capabilities, vehicular traffic congestion levels are significantly affected so that nodes' connectivity is also affected.

3. Overview of the simulation tools

In this section, we provide some details about the SUMO simulator adopted for the analysis. From a general perspective, the essential elements for a vehicular simulation include: (i) the road network; and (ii) the traffic demand. First, the *road network* refers to the traffic-related part of a map, including roads (edges), traffic lights, intersections, and the logic between elements. Second, *traffic demand* refers to the number of vehicles circulating in the road network at a given time. This last also defines the type of vehicles and their correspondent set of roads to follow, known as the vehicle's route. Here, we focus on the traffic demand aspect, and on the SUMO build-in tools intended for the traffic demand generation. Also, we will introduce the reroute functionality that enables dynamic route planning. In this case, vehicles can dynamically modify their route during the simulation emulating the user behavior in case of route events (e.g., traffic jams).

3.1. SUMO overview

The first step is to build the road network. By using SUMO tools, two methods can be followed: (i) create a synthetic network (e.g., a Manhattan type network), or (ii) generate the network based on real maps. In this work, we follow the second approach and import a map from OSM repositories [16]. Here, OSM excels other repositories because it is free and presents an excellent real-world representation. An overview of the used tools is as follows:

Netconvert: It takes digital map files from different sources (e.g., OSM, VISSIM, MATsim) and converts them into one road network file readable by SUMO, called *.net* file. It provides a full set of processing options for map conversion. A complete list of options is available in [18]. Common options to characterize urban environment maps, are *remove-edges.by-vclass, remove-edges.isolated, no-turnarounds, and geometry.remove*.

Polyconvert: It allows us to import geometrical shapes (polygons) from different sources (e.g., OSM). It is particularly important when evaluating vehicular networks operating in suburban and urban environments. Due to the presence of buildings (i.e., polygons), radio transmissions are heavily impacted by signal shadowing effects in this environment.

Once the road network has been defined, to further modify or debug network properties (e.g., roads' length, traffic lights, intersections logic), the SUMO package includes a useful graphical tool called *Netedit*. It is a visual network editor that allows us to edit/create all available network properties and infrastructure in the network (e.g., bus stops, traffic light cycles, between others).

Then, the traffic demand generation includes the vehicle type that describes the vehicle's physical properties (e.g., size, gas/electric fueled), a route the vehicle shall take, and the vehicle itself. A single route can be assigned to a single-vehicle (e.g., trips, routes) or several vehicles (e.g., flows of vehicles). In SUMO, routes can be defined in three different ways: (i) manually, specifying all the components of the route (i.e., set of edges), (ii) generating routes randomly, or (iii) using the SUMO generation tools. In this work, we use (ii) and (iii) to randomly create routes (RandomTrips) for the road network and using the SUMO generation tools (OD2Trips, DUARouter, DUAIterate, MARouter) [19]. In the next subsection, each of these tools are explained in detail.

3.2. Traffic demand generation tools

The SUMO simulator includes, by default, some facilitates to generate the vehicular traffic demand. Here, we present details of the commonly used traffic generation tools: OD2trips, DUArouter, DUAIterate, MArouter, and RandomTrips; see Table 2. First, input files required for the generation of traffic and output files defining the form of vehicle traces (e.g., routes, trips). The traffic assignment method refers to the mechanism used to compute the route for each vehicle. By default, SUMO describes three traffic assignment methods, (i) Empty-network, (ii) incremental, and (iii) iterative. In the (i) case, a.k.a naive user assignment, vehicles compute their routes under the assumption that they are alone in the network (empty-network). In contrast to (i), in the (ii) case, a.k.a. user assignment, vehi-

Table 2: SUMO traffic generation tools.

Tool	Inputs	Outputs	Traffic assignment method
OD2Trips (OD2)	Network, O/D Matrices, Additionals	Trips	Incremental
DUARouter (DUAR)	Network, Trips, Additionals	Routes	Empty-network
MARouter (MAR)	Network, O/D Matrices, Additionals	Flows	Incremental
RandomTrips (RT)	Network, Additionals	Trips	Incremental
DUAIterate (DUAI)	Network, Trips, Additionals	Routes	Iterative

cles compute their routes at the time of departure considering all the vehicles currently running in the simulation (i.e., traffic-loaded network). The (iii) case, a.k.a user equilibrium, tries to select the best route (e.g., fastest route) for each vehicle in the simulation. In the following, the main configurations for tools in Table 2 are described:

OD2Trips (OD2): The inputs consist of the road network, traffic analysis zones, O/D matrices (i.e., origin/destination districts), and vehicle types definitions coded as additional entries, see Table 2. As the output, OD2 generates a list of trips definitions according to the number of vehicles to be inserted within the time interval coded in the O/D matrices. To allocate the traffic demand in the network, OD2 uses an incremental assignment method, see Table 2. Here, vehicles will calculate fastest paths according to their time of departure considering a traffic-loaded network. In this way, the incremental assignment prevents all vehicles from choosing the same route (i.e., prevents congested routes). By default, vehicles are uniformly distributed within the time interval coded in matrices. Although, OD2 includes a daily time lines option, which allows to split an O/D-matrix over a day (24 hours) with a granularity of 1 hour. In Section 4.5, we configure an O/D matrix for each time interval of 15 minutes increasing the granularity of the traffic generation to 1/4 hour.

DUARouter (DUAR): In addition to the road network and additional files (e.g., vehicle types), the DUAR tool imports different demand definitions (trips or flows). To obtain the same traffic pattern previously coded in O/D matrices, here we configure the OD2 output (trips) as the input of the DUARouter tool, see Table 2. In contrast to the OD2 tool, the DUARouter output consists of a list of routes that includes the full path between the origin and destination points, see Table 2. DUAR computes the shortest path between origin and destination points regarding to the trip time or the trip length. For this, some routing algorithms can be configured (e.g., Astar, Dijkstra). By default, DUAR uses a modified version of the Dijkstra algorithm [20]. Besides, with DUAR, each vehicle computes its route under the assumption that they are alone in the network. The traffic assignation method follows an empty-network assignation, see Table 2. Depending on the amount of generated traffic, certain roads in the network can be rapidly congested (e.g., highways).

MARouter (MAR): The MARouter tool is able to compute a microscopic (flows) or macroscopic user assignment. As with OD2, MAR inputs include, O/D matrices, and vehicle definitions. Here, we use the same O/D matrices used with the OD2 tool. As the output, the MAR generates a list of vehicle flows (i.e., microscopic), including the route distributions meaning that each route includes the probability to be selected. Here, the traffic assignation method employs resistive functions that approximate the travel time increment when the number of vehicles in the flow increases. In this manner, in contrast to the DUAR tool, the MAR tool generates routes considering a traffic-loaded network.

RandomTrips (RT): This tool allows researchers to quickly generate a set of random trips within a time interval. The inputs consist of the road network, simulation begin/end time, vehicles definition, and vehicles' arrival rate. This last parameter is defined by $(t_1 - t_0)/n$, where n represents the number of vehicles to be inserted between times t_0 and t_1. By default, source and destination points are randomly selected. The output file consist of a list of trips with source and destination edges. With RT tool, vehicle traces are computed at the time of departure in the simulation, know as an incremental assignment, see Table 2. As with the OD2 tool, the RT tool prevents bottlenecks in the network. Finally, as origin and destination points are randomly selected, those may not have a valid connection in the road network.

DUAIterate (DUAI): As with DUAR, the DUAI tool uses the road network, trips or routes, and additional files as inputs for the tool, see Table 2. It uses an assignment method called iterative, see Table 2, which tries to calculate the user equilibrium, meaning that it generate a route for each vehicle where the route cost (e.g., travel time) cannot be reduced by using an alternative route. This is done by iteratively calling the DUAR tool. Routes generation process is as follows:

1. Call DUAR in order to generate a shortest route for each vehicle using an empty-network assignment, see Table 2.
2. Run the simulation using the set of routes generated in the previous step. The idea is to obtain the costs of current routes to be used in the next iteration.
3. Compare current mean route costs (i.e., mean travel times) with its last value known. In case of the mean travel time decreases below a given threshold, it is said that the algorithm converges finishing the execution.
4. In case the mean travel time is above the given threshold, new routes are computed for each vehicle using the last known routes' costs. New routes are aggregated to the correspondent vehicle's set. Then, vehicles' routes are collected within a route distribution and used to choose the route to drive in the next iteration. At this point, the route choosing mechanism (Gawron or Logit algorithms) is in

Table 3: Comparison of the main features of SUMO traffic demand generation tools. O/D means (Origen/Destination)

	Characteristics	OD2	DUAR	MAR	RT	DUAI
Routes Generation	Use the Dijkstra algorithm	✗	✓	✗	✗	✓
	Use detectors	✓	✗	✗	✗	✗
	Use O/D matrices	✓	✗	✓	✗	✗
	O/D points are mandatory	✓	✓	✓	✗	✓
	Full traffic files	✗	✓	✓	✗	✓
Traffic Demand Support	> 3 vehicle types	✓	✓	✓	✓	✓
	Buss stops for vehicle flows	✗	✓	✗	✓	✓
	TAZ districts	✓	✗	✓	✗	✗
Efficient Scenarios	Urban areas	✓	✓	✓	✓	✓
	Rural areas	✗	✓	✗	✓	✓
Avoid	Congestion situations	✓	✗	✓	✓	✓
	Deadlock network	✓	✗	✗	✓	✓

charge to select the vehicle's route from the set of alternative routes.

3.3. Comparison of SUMO re-routing tools

In Table 3 we present a comparison of the SUMO re-routing tools introduced in the previous section. We classify tools regarding their route generation features, traffic features, efficiency scenarios, and their capacity to prevent/avoid congestion and deadlock network situations. The main characteristics of the SUMO re-routing tools are summarized as follows:

- The tool that requires fewer configuration files is RT. OD2 and MAR require O/D matrices, while DUAR and DUAI require trip files. Notice that O/D matrices usually refer to a traffic assignment zones (TAZ) file. Finally, the network file is mandatory in all cases.
- OD2 and MAR tools support TAZ. This facilitates the traffic demand generation between districts.
- OD2 and MAR allows us to generate traffic demand per hour (Veh/hr defined in the O/D matrix file), while RT uniformly distributes traffic demand within a time interval. DUA and DUAI support per hour traffic generation because these tools can use trip files as inputs.
- DUAR, DUAI, and MAR traffic demand files include the whole vehicle's path (i.e., set of edges) while OD2 and RT only have O/D points. For this reason, in the first case, traffic files are more extensive. Notice that OD2 and RT compute vehicles' routes at vehicles' departure times.
- OD2, MAR RT, are intended to avoid congestion situations (refer to Section 3.2). In contrast, the DUAR tool may generate congested roads and deadlock networks. Lastly, DUAI avoids traffic congestion situations, but it requires a high amount of resources to process iterations as it is described later in Section 4.5.
- All the tools support >3 types of vehicles simultaneously.
- For all route generation tools, at least one origin point must be defined, while the destination points are mandatory for DUAR, DUAI, OD2, and MAR.

3.4. Dynamic route planning

Nowadays, most modern vehicles are equipped with real-time navigation services, allowing drivers to plan their routes before and during their trip. In this context, to provide realism to simulations, vehicles should update their routes during their journeys.

By default, in SUMO vehicles will follow their corresponding route during the simulation without modifying it. Following this approach, routes are static and vehicles do not update it. Under this situation, if a road becomes congested vehicles will not modify their routes as it would probably happen in a realistic scenario (according to the expected driver behavior). To cope with this issue, vehicles' re-routing capabilities can be enabled. This will allow vehicles to update their routes during the simulation [21] to avoid a congestion situation. This approach works by giving vehicles the ability to re-compute their routes periodically (i.e., re-routing is triggered by the vehicle). For this, some or all the vehicles in the simulation can be equipped with a re-routing device [22]. In SUMO, by default the percentage of vehicles that have a re-routing device is set to 0%. Four main steps are considered in order to configure the vehicles' route updating during the simulation:

Step 1: Before the simulation initializes, each vehicle has assigned a vehicle type and a route, as it is detailed in Section 3. Also, the percentage of vehicles equipped with a re-routing device is defined.

Step 2: Once the simulation starts, conditions along the road network are updated following a fixed time interval of t seconds. For this, SUMO assigns weights to each edge in the network. The evolution of the edges' weights during simulation is computed as an estimation of the travel time (time required to traverse an edge). The travel time is estimated by calculating the mean speed on the road and then dividing the road length by that speed.

Step 3: Vehicles equipped with a re-routing device re-apply the optimal path computation following the r (seconds) re-routing interval. r refers to the interval time elapsed between consecutive re-routing events. If the travel time on the new route is shorter than that the current one, the vehicle changes the route accordingly.

Step 4: At this time, to select the fastest path for the current vehicle, SUMO uses the vehicle's current position, destination's position, and current status in the road network (updated at t time). If the new path results faster than the current one, the vehicle updates its route accordingly.

4. Traffic mobility evaluation

In this section, we perform an in-depth review of the SUMO traffic generation tools listed in Table 2. This evaluation aims to present to SUMO users a clear view of the fundamental differences between available re-routing tools. To this end, we perform a set of analysis as follows:

1. Traffic intensity measurement: we register the number of vehicles generated by each tool and compare it against real

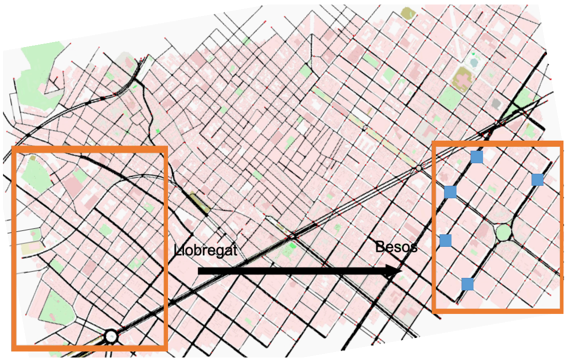


Figure 1: Barcelona, Spain. Gracia district. Mid-size road network ($7 km^2$). Map taken from OpenStreetMaps [16].

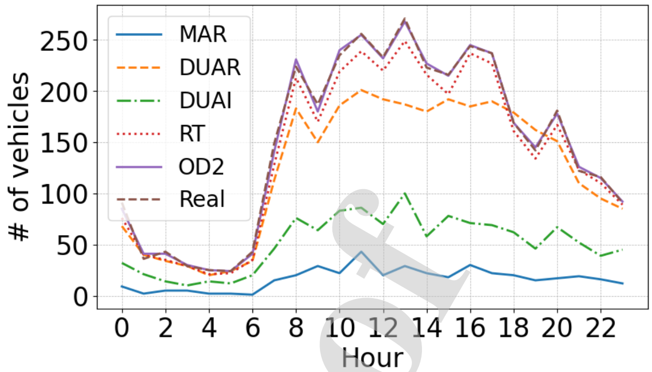


Figure 2: Traffic intensity during a working day in Travessera de Gracia street, Barcelona, Spain. Traffic is measured each 15 minutes in the direction Llobregat to Besos. Data extracted from [5].

city traffic statistics. This gives researchers a clear view of the accuracy of the traffic demand generated by SUMO.

2. Re-route conditions: utilizing trip times, trip length, and route emissions show the route conditions generated by each tool (e.g., which tool produces the highest emissions).
3. Resources usage: This analysis intends to provide a clear idea of the minimum amount of resources (CPU, memory and disk usage) required to execute each re-routing tool.

In this sense, a SUMO user may require a tool that provides a realistic traffic demand where the routes of vehicles produce the lowest emissions when they are executed in dedicated hardware with limited resources.

4.1. Simulation scenario

In this section, we detail the configuration of a simulation scenario, as an example. First, we generate the road network using a real map imported from OSM [16]. Then, through the use of the *Netconvert* tool, we generate the .ned file, which is understandable for the SUMO simulator, as described in Section 3.1. Fig. 1 shows the area of study, which corresponds to the Gracia district in Barcelona, Spain. We consider an area of $7\ km^2$. Exclusive lanes for the tram, rail, electric rail, bicycles, and pedestrians have been removed from the road network due to the road map tuning process. Besides, buildings information has been generated using the *Polyconvert* tool described in Section 3.1. Finally, the study area has been tuned using the *Netedit* tool (e.g., maximum speed of the roads).

Once the road network has been defined, to generate the traffic demand we consider the real traffic according to the Barcelona's City Hall [5] regarding the vehicles' flow traveling in the considered area going in the Llobregat to Besos direction, see Fig. 1. We generate full-day traffic (i.e., 24 hours) according to the real traffic measured during a working day in [5]. For this, we convert the traffic analysis zones (Llobregat as the origin and Besos as the destination) into traffic assignment zones, called TAZs, see Fig. 1. A TAZ is defined by an identifier (e.g., zone name) and the list of roads, a.k.a. edges, within the analysis zone. Then, we code the number of vehicles that drive from the origin (O) zone to the destination (D) zones in O/D matrices. The O/D matrices contain information about the number of vehicles and the source/destination TAZs. Later, the generated O/D matrices are used as inputs to the SUMO traffic generation tools in Table 2, as detailed in Section 3.2. As the outputs of the generation tools, we can obtain a list of vehicle trips (including the O/D edges), a list of routes (including the full trace), or a list of flows (including O/D edges or the full trace), see Table 2.

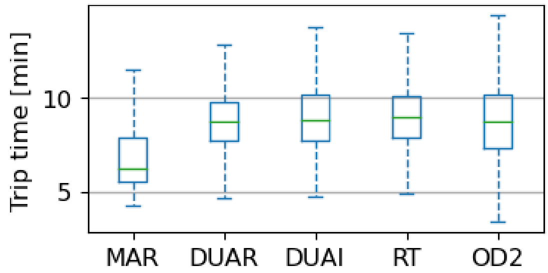
Finally, we have implemented the re-routing feature using a SUMO device, as detailed in Section 3.4. A vehicle equipped with a re-routing device will update its route following a re-routing interval of *t* seconds. For this, we consider the typical GPS update frequency of 1Hz.

In the following sections, we perform a thorough analysis of the different SUMO traffic generation tools and the dynamic route planning of vehicles. For this, we evaluate generated traffic demand in terms of the traffic intensity, fundamental mobility measures (trip time/distance), and the total emissions (NOx emissions) during the simulation. Additionally, we consider vehicles' re-route capabilities and their impact in the overall mobility. Finally, we evaluate the performance of SUMO tools in terms of resource usage (i.e. CPU, RAM and disk usage). The goal is to assess the impact of tools as well as the vehicles' re-route capabilities on traffic mobility, in Section 4 and Section 4.4. Finally, we evaluate the impact of the vehicles' re-route capability on the nodes' connectivity in Section 5.

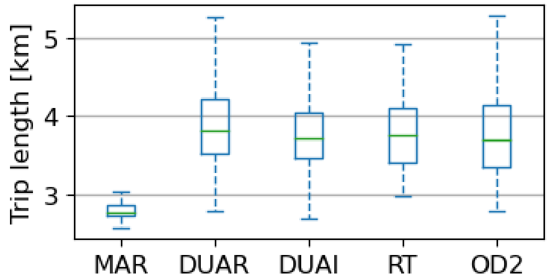
4.2. Traffic intensity

In Fig. 2, we can see the traffic traversing the Gracia district in Barcelona, Spain, in the Llobregat to Besos direction, see Fig. 1. Here, the real case in dashed brown color depicts the real traffic in the Gracia district obtained from the Barcelona city statistics in [5]. The set of traffic demands have been generated using our proposed STG tool, which is based on real induction loop measurements taken from [5]. Details of the STG tool are included in Section 6.

In Fig. 2 we can see that the OD2 and RT tools generate the closest traffic intensity to the real case. This is because OD2 and RT use an incremental traffic assignment mechanism. In both



(a)



(b)

Figure 3: (a) Mean trip time and (b) mean trip length measured for the different routes generated with the SUMO tools listed in Table 2.

cases, routes are computed according to the vehicle's departure time and considering a traffic-loaded network. In this manner, vehicles can be allocated at the designated time (i.e., time of the day) without overloading roads along the map. In contrast, the DUAR tool presents similar results when compared with the real case, just below the RT tool, see Fig. 2. We can see that the DUAR tool's peak hour is reached at 11 am, which differs from the real traffic behavior (peak-hour reached at 13 pm). This is due to DUAR uses and empty-network traffic assignment mechanism, where it is more probable that vehicles choose the same route during the simulation.

Besides, Fig. 2 shows that DUAI and MAR present distant results from the actual traffic demand. In the first case, DUAR tool looks for the user equilibrium meaning that each vehicle will have the route with the minimum cost (e.g., travel time) considering a traffic-loaded network. Although DUAR shows a similar traffic pattern compared to the real traffic, alternative routes in the evaluated zone (i.e., traffic detectors) are selected. In the same way, MAR tool selects routes with the lowest costs considering the whole set of vehicles in the route (i.e., flows). Although MAR uses an incremental traffic assignment mechanism, it also generates flows of vehicles outside the evaluated zone.

Notice that even though the same number of vehicles is considered as inputs for all the traffic generation tools, DUAI and MAR tools show a quite lower number of vehicles measured during the simulation.

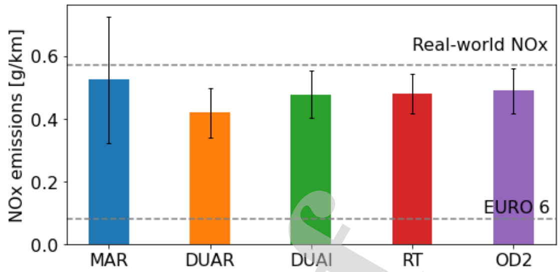


Figure 4: Total nitrogen oxide (NOx) emissions in the considered area during a full-day.

4.3. *Traffic mobility measures*

This section evaluates the SUMO routing tools using fundamental traffic measures: (i) mean trip time and (ii) mean trip length. (i) the mean trip time is defined as the mean time required for the driver to travel from origin to destination. (ii) the mean trip length is defined as the mean route length from source to destination. Also, we evaluate the total amount of emissions produced by vehicles during a full-day simulation. This last metric allows us to examine generated routes in terms of contamination through the simulation area. For this, we compare generated emissions against the real-world emissions in [8].

Mean trip time: Fig. 3a shows the mean trip time of routes generated with the SUMO routing tools listed in Table 2. On the one hand, the MAR tool attains the lowest trip time (7 minutes). This is because the MAR tool selects routes with the closest source/destination points so that it generates the shortest routes, as we will see in the next section. The MAR tool may be useful in the case of traffic generation considering several districts (i.e., different traffic analysis zones). On the other hand, the DUAI tool shows the highest value (9 minutes). In this case, the shortest route is not guaranteed as DUAI aims to avoid congested situations. Although DUAR and DUAI present similar results, the DUAI tool results more adequate in case of crowded scenarios (e.g., thousands of vehicles). Finally, we can see that within the OD2 tool, trip time values present a disperse distribution. This is because origin/destination points are randomly distributed. Besides, the incremental assignment mechanism selects different routes for all the vehicles looking to avoid congested roads.

Mean trip length: Fig. 3b shows the mean trip length of routes generated with the SUMO routing tools listed in Table 2. On the one hand, MAR attains the lowest traveled distance (2.8 km). Besides, we can see that with the MAR tool the travel distances present a skewed distribution. This is mainly due to MAR tool generate flows of vehicles where all the vehicles that compose the flow follow the same route. This corresponds to what was seen in Fig. 3a, where MAR reaches the lowest mean trip time. On the other hand, DUAR attains the highest traveled distance (3.9 km). Besides, DUAI and RT obtain similar travel distances (3.7 km) just below the DUAR tool.

Total emissions: To evaluate the vehicles' emissions, we consider Nitrogen oxides (NOx) coming from the reaction of nitrogen and oxygen in heat combustion engines. NOx emissions are in recent years tightly regulated due to its dangerous effects on human health. We consider the same scenario depicted in Fig. 1. Here, the type of vehicle corresponds to diesel passenger cars, since the majority of passenger cars in Barcelona are diesel (60%) according to [23]. Besides, we define a speed profile with $30-50\ km/h$ speed values uniformly distributed among vehicles in movement. We use the SUMO emission model based on the Handbook Emission Factors for Road Transport version 3 (HBEFA3) [24]. The vehicle configuration regarding emissions are summarized in Table 4.

The main elements to consider in an emissions model include vehicles' acceleration, speed, and road slope. The emissions model in [24], is implemented by fitting data extracted from the HBEFA dataset. Here, the model is considered as a linear problem where the resulting function's coefficients denote the different emission classes (e.g., light-duty vehicles). In this sense, the model lacks the dependency on the vehicles' acceleration, speed, or roads' slope. As a result, the obtained values do not match some basic emission properties (e.g., emissions are always above zero, or vehicles produce emissions when velocity is 0 m/s). Besides, since the EURO 5 norm, most of the new vehicles are equipped with a fuel-saving technology so-called start-stop [25]. This system reduces emissions by cutting the engine when the car comes to a complete stop situation (e.g., traffic light or traffic jam).

To measure the total NOx emissions, current SUMO model does not consider traffic jam situations (i.e., $speed = 0$ for an extended period of time). In such case, vehicle's emissions would get infinite emissions using the current SUMO model. However, this is not a realistic assumption, as we have commented above. To avoid this misbehavior of the NOx emissions model, in Algorithm 1 we set an upper limit for the emissions of the SUMO's HBEFA3-based vehicle. First, in order to determine whether the vehicle is in a traffic light stop or in a congestion situation (i.e., in a traffic jam), an auxiliary timer for vehicle n named Car^n_{timer} is compared against the traffic light cycle time (TLC), see line 4 in Algorithm 1. The TLC includes the traffic light yellow-red-green cycle. In case the Car^n_{timer} exceeds the TLC, we consider that vehicle n experiences a congestion situation and then the Car^n_{NOx} is set to 0 (i.e., the vehicle is not generating NOx emissions). In the other case, the $Car^n_{NOx_t}$ is saved and the Car^n_{timer} increments, see lines 8 and 9, respectively. Finally, the total Car^n_{NOx} is reported at line 12.

To evaluate the simulated NOx emissions results, we compare the values obtained with SUMO improved with our Algorithm 1, with values obtained from measures of the real-world NOx emissions in Barcelona [8]. Additionally, we also compare the results to the EURO 6 norm [7], which states that all newly diesel passenger cars must meet a NOx emission limit of $0.080\ g/km$. In Fig. 4, we can see the total NOx emissions obtained by the evaluated SUMO traffic generation tools together with our Algorithm 1. On the one hand, we can see that for all the SUMO tools (listed in Table 2) the results largely exceed the EURO 6 norm emissions limit defined for diesel vehicles. This is mainly due to the fact that the EURO 6 standard measures emission limits under a controlled environment (i.e., not a real-wold test). On the other hand, the SUMO routing tools reach values more proximate to real-world driving conditions. Here, the MAR tool overpass the real-world NOx emissions and reaches higher values compared to the other tools. This is mainly because the MAR tool generates vehicles flows as outputs, see Table 2. Under this scheme, the set of vehicles in the same flow depends on the speed of the vehicles at the head of the flow. Here, vehicles at the queue increase their travel times (see Fig. 3a) and therefore the NOx emissions.

Finally, we can conclude that by including Algorithm 1 in the current SUMO emissions model, realistic NOx emission values can be achieved with SUMO tools.

Table 4: Vehicle configuration to evaluate the emissions.

Type of vehicle	Speed	Emissions model	Emissions class
Diesel passenger car	(30-50) km/hr	HBEFA3-based [24]	LDV_D_EU6

Algorithm 1 Upper limit for the car emissions model HBEFA3 in SUMO

1: $Car^n_{NOx_t}$ = Car n NOx emissions at time t
2: $Car^n_{s_t}$ = Car n speed at time t
3: **while** $Car^n_{speed} == 0$ **do**
4:     **if** $Car^n_{timer} \geq$ TLC **then**
5:         $Car^n_{NOx} = 0$
6:         break
7:     **else**
8:         Car^n_{NOx} = previous $Car^n_{NOx} + Car^n_{NOx_t}$
9:         update Car^n_{timer}
10:     **end if**
11: **end while**
12: report Car^n_{NOx}

4.4. Re-route capabilities

In SUMO, each vehicle can be equipped with a re-routing device, allowing vehicles to update their routes during their trips. A re-routing device refers to a container for data and functionality that enables vehicles to interact with their environment. In this context, to attain a realistic simulation that considers the user behavior the user may decide to change its route based on current road conditions (e.g., there is a congested route). Here, we equip all the vehicles in the simulation with SUMO re-routing devices. For this, we consider that the on-board GPS module updates at a rate of 1Hz, meaning that each second the status of roads will be updated. We consider two scenarios:

- (i) without re-routing capabilities: None of the vehicles can change its route (default SUMO configuration).
- (ii) with re-routing capabilities: All the vehicles in the simulation can update their routes during their trips.

Fig. 5 shows vehicles' density (veh/km) in each road of the network during a full-day simulation. On the left side of Fig. 5 we have the results for case (i); and on the right side, the results for case (ii). We show simulation results for all the traffic generation tools listed in Table 2.

In Fig. 5a and Fig. 5b, we can see that for (i) and (ii) cases with the OD2 tool vehicles follow the same routes to go from origin to destination zones (i.e., from Llobregat to Besos). In this case, re-routing capabilities do not modify the traffic mobility. This is because the OD2 tool uses an incremental traffic assignment method, where vehicles compute their routes at the moment of departure considering a traffic-loaded network.

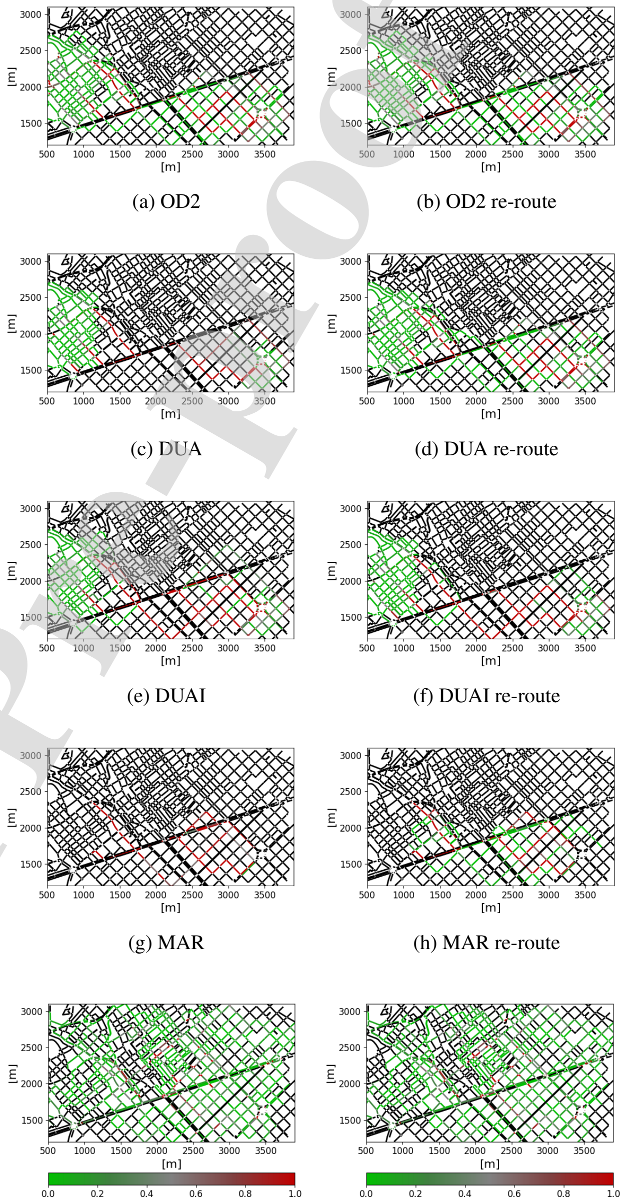
The opposite case occurs with the DUAR tool in Fig. 5c and Fig. 5d. In Fig. 5c we have case (i), we can see that vehicles mainly select the same routes, see red lines in Fig. 5c. In Fig. 5d we have case (ii), we can see that alternative routes are selected from origin to destination, see green lines in Fig. 5d. With the DUAR tool, vehicle routes are assigned using the empty-network method (i.e., vehicles assume they are alone in the network), and therefore the fastest routes are selected. In case (ii), even though vehicles have chosen the same routes before departure, they update their routes dynamically during the simulation. Here, vehicles select alternative routes as soon as they receive road status updates.

In Fig. 5e and Fig. 5d, the DUAI tool shows similar traffic mobility for both cases (i) and (ii). This is because the DUAI tool looks for each vehicle's best possible route considering a traffic-loaded network. This is done by iteratively executing the DUAR tool and using statistical information in the next iteration. As well as with the OD2 tool, the DUAI tool has no impact on the generated traffic mobility.

Regarding the MAR tool, in Fig. 5g and Fig. 5h we show cases (i) and (ii), respectively. First, in Fig. 5g, we can see that vehicles mainly select the same route to reach destination points. This is because the MAR tool generates vehicles (i.e., a set of vehicles with the same traces) where the closest routes between origin and destination zones are selected. In Fig. 5h case (ii) shows that alternative routes are selected to reach destination points, with a similar behavior to what we see with the DUAR tool.

Finally, Fig. 5i and Fig. 5j shows the RT tool for cases (i) and (ii), respectively. We can see that the traffic mobility remains the same for both cases. As with the OD2 tool, the RT tool uses the incremental traffic assignment method to update their routes before departure.

Besides, in Fig. 5, we can see that for both (i) and (ii) cases and for the OD2, DUAR and DUAI tools, origin and destination points are randomly distributed within source and destination zones (i.e., Llobregat and Besos). This is due to the fact that OD2 receives as input the O/D matrices based on TAZs. Similarly, DUAR and DUAI use OD2 files as inputs of these tools and therefore produce a similar origin-destination location. In contrast, the MAR and RT tools do not follow the traffic assignment zones (i.e., Llobregat and Besos) to locate origin and destination points. Even though MAR receives as input the O/D matrices, it selects the origin and destination points situated in the TAZ border and with the minimum distance from source to destination. In the case of the RT tool, it locates origin and destination points throughout the entire map. Regarding the RT tool, in Fig. 5i we can see that this approach shows a uniform road's usage when compared with other tools.



(a) OD2 (b) OD2 re-route

(c) DUA (d) DUA re-route

(e) DUAI (f) DUAI re-route

(g) MAR (h) MAR re-route

(i) RT (j) RT re-route

Figure 5: Vehicles density [veh/km] during a full-day simulation. On the left side, figures show no re-route capabilities. On the right side, figures show re-route capabilities enabled.

4.5. Resource usage

In this section, we assess the SUMO traffic generation tools in terms of CPU, memory and disk usage, see Fig. 6. For this, we consider the whole simulation process including the time needed to generate the traffic and the simulation time.

Fig. 6a shows the % of CPU usage. Here, DUAR, DUAI and RT tools generate peaks of use. This is because these tools support multi-threats processing (i.e., vehicles' routing is executed in parallel). The DUAI tool attains the highest computation time (> 4 minutes). This is because DUAI conducts iterative simulations to avoid congested routes. In contrast, the OD2 tool shows a lower computation time. It takes about 1 minute and 13% of CPU usage to generate the traffic and execute the simulation.

Fig. 6b shows the % of memory (RAM) usage. To generate a full day of traffic (24 hours) with a resolution of 15 minutes (4 times in an hour), the RT is executed 96 times. This last takes around 1 minute before starting the simulation. In the same way, MAR requires near to 2 minutes to generate the traffic demand before starting the simulation. It attains the highest memory consumption during this period. This is because MAR considers a classic macroscopic traffic assignment where the correspondent route distribution for each flow is handled in memory to estimate later the route cost (i.e., estimate the travel time). On the other hand, DUAR, DUAI, and OD2 tools use O/D matrices or trips as inputs, which require a lower traffic generation time. Here, DUAI achieves the lower memory consumption ($\approx$ 1% of memory).

Finally, to measure the disk space usage needed by each tool, we consider traffic generation files and simulation output files. Traffic generation files are required to generate the traffic demand (e.g., O/D matrices, trips, routes, alternative routes, route' cost). Simulation output files are used to get statistics from simulations (e.g., tripinfo, summary, emissions, floating car data). Fig. 6c shows the disk space usage for each tool. Here, we can see that the DUAI tool requires the highest disk space (near 3GB). This is because it generates a set of additional files per iteration. In contrast to DUAR, that uses high levels of memory, DUAI writes files to disk. Compared to MAR and OD2, DUAI requires three times more disk space. While DUAR and RT need almost the same amount of disk, when comparing with MAR and OD2 tools they require twice more disk space.

We can see that with a mid-size road network (see Fig. 1), the DUAI requires four times the computation time when comparing to the OD2 tool (see Fig. 6a). DUAR, MAR, and RT tools show similar computation times (between 3-4 minutes), see Fig. 6a. Besides, the DUAI reaches the lowest memory usage while MAR, DUAR, and OD2 attain the highest memory usage near to 4% of the total memory, see Fig. 6b. Regarding disk usage, DUAI shows the highest requirements, see Fig. 6c. Here, in large-size scenarios (e.g., large road network size with an increased number of vehicles), DUAI may be infeasible because it requires a considerable amount of disk space.

Simulations were executed in the same computer equipped with an Intel Xeon CPU E5-2637 at 3.50GHz and 16 GB of RAM. The operating system is Ubuntu Linux 18.04 LTS with a 64-bit kernel. We used the latest SUMO version 1.8.0.

4.6. Discussion

Regarding the traffic intensity, we can see that the OD2 and RT tools generate the traffic intensity which is closest compared to the real-world traffic (see Fig. 2). In contrast, the MAR and DUAI tools generate a traffic intensity just below the RT tool. Besides, the DUAR tool generates a traffic demand that differs from real-world traffic (out-of-time peak hour) mainly due to congestion situations during simulations (see Fig. 2).

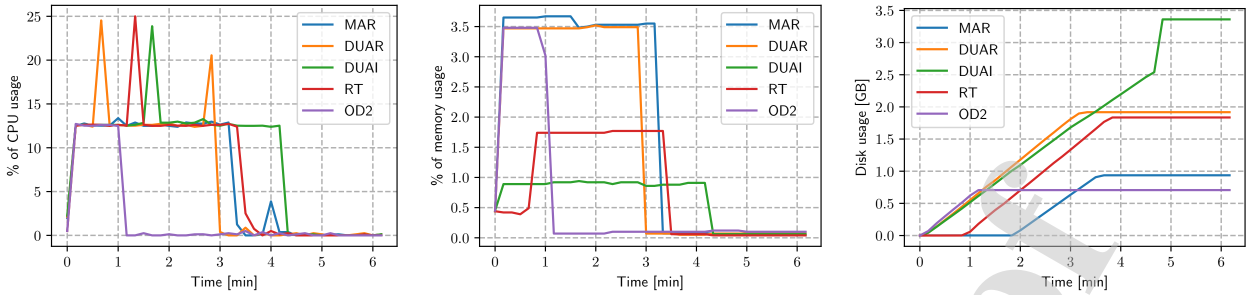
Concerning the mobility measures, we can see that the MAR tool generates traces with the lowest mean trip time and distance, while the DUAI tool presents the highest values (see Fig. 3). We can see that the DUAI tool will result adequately in crowded scenarios (e.g., thousands of vehicles), where DUAR generates deadlock networks. In this sense, to cope with the network deadlock problem, re-routing capabilities allow vehicles to select alternative routes to those generated before departure. Besides, using Algorithm 1 and the SUMO emissions model (HBEFA3-based), NOx emissions reach values near to the real-work measures ($< 0.6g/km$) and above the EURO 6 norm for diesel passenger cars ($> 0.080g/km$) (see Fig. 4).

Finally, in terms of resource usage, we can see that the DUAI tool requires four times the CPU usage when comparing to the OD2 tool (see Fig. 6a). DUAR, MAR and RT tools show similar computation times, i.e. between 3-4 minutes (see Fig. 6a). MAR, DUAR, and OD2 attain the highest memory usage (see Fig. 6b). In contrast, the DUAI tool achieves lower memory consumption, even though it requires higher disk space (see Fig. 6c).

5. Performance evaluation of the network connectivity

To provide an idea of the impact of vehicles' re-routing capabilities on the network connectivity, we take advantage of graph theory concepts. First, we visually evaluate the nodes' connectivity through an adjacency matrix. Then, through the use of well-know graph metrics [26], we intend to provide valuable insight into the vehicular network behavior. To this end, we consider a higher traffic demand for the hole district of Gracia. Notice that in Section 4 we uniquely consider the traffic going from Llobregat to Besos direction. Here, the traffic demand is generated according to the Barcelona's City Hall statistics in [6]. For the Gracia district, in rush hours there are around 17,500 veh/hour. Then, we have derived the number of vehicles for 40%, 70%, and 100% of the maximum real traffic demand; henceforth low, medium, and high vehicular traffic demand, respectively. We have generated vehicles for three hours and for each traffic demand using the RT traffic generation tool, as it shows a traffic demand more uniformly distributed in Section 4.4 and a good performance, according to Fig. 6. Also, we include the re-routing functionality of SUMO. For this, we consider the following scenarios:

- NR: None of the vehicles is capable of changing its route (default SUMO configuration).
- R - 1 min: All the vehicles in the simulation can change their routes following a re-routing interval of r = 1 min.



(a) (b) (c)

Figure 6: Comparison of SUMO re-routing tools in terms of (a) CPU usage, (b) RAM usage, and (c) disk usage. Simulation time includes O/D matrices generation time and simulation execution time. (c) includes commonly used SUMO output files (FCD, emissions, tripinfo).

- R - 3 min: All the vehicles in the simulation can change their routes following a re-routing interval of $r = 3$ min.
- R - 6 min: All the vehicles in the simulation can change their routes following a re-routing interval of $r = 6$ min.

Table 5: Simulation settings of the vehicular network, see Fig. 1.

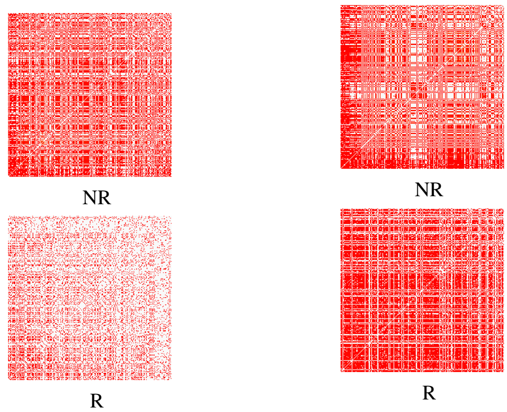
Parameters	Value
Map area	7 km^2
Low traffic flow	7,000 veh/h
Medium traffic flow	12,250 veh/h
High traffic flow	17,500 veh/h
Transmission power	23 dbm
Sensitivity threshold	-82 dbm
Transmission range	400 m in line of sight (LOS)
Path loss model	Empirical IEEE 802.11p [27]

The main vehicular network simulation settings are summarized in Table 5. Here, we assume all vehicles in the scenario are equipped with an IEEE 802.11p network card and have defined a transmission power of 23 dBm. Thus, the communication range will be approximately 400m. We also consider the building attenuation model proposed in [27] to model the non-line of sight (NLOS) condition to generate the building information, as it is described in Section 3.1. The simulation area corresponds to Fig. 1, with an area of 7 km^2. In our case, we are interested in analyzing the impact that the flow of vehicles through the streets has on the vehicular network connectivity. We vary the vehicles' flow from a low traffic flow (7,000 veh/h), a medium traffic flow (12,250 veh/h), and a high traffic flow (17,500 veh/h).

The first step to analyze the network is to convert vehicular traces in network snapshots at different simulation times. Then, the nodes' connectivity can be represented as a graph G where $V(G)$ refers to the set of *vertices* (nodes) of the graph, and $E(G)$ is the set of *edges* of the graph. An edge is added between two nodes (i, j) in case those nodes reach each other (i.e., reception power > -82 dBm). The network is represented by the adjacency matrix ($A_{i,j}$), where each entry corresponds to the weight of the edge ($E_{w_{i,j}}$). This weight of the edge corresponds to the signal strength in the link between two nodes, as follows:

$$A_{i,j} = \begin{cases} E_{w_{i,j}} & \text{if nodes } i, j \text{ are connected} \\ 0 & \textit{otherwise} \end{cases} \quad (1)$$

In Fig. 7, we depict the nodes' connectivity by means of the adjacency matrix. To obtain the edges' weight of the graph, we have added the signal strength each time that two nodes are in contact during the period of study. We take into account the 250 nodes with the highest signal strength. The connection intensity varies from light-red (few nodes are connected) to very dark-red (many nodes are connected), depending on the node's edge weights. We evaluate two cases considering low (off-peak hours) and high (peak hours) traffic demands. In the figure, *R* cases mean that all vehicles in the simulation can perform route changes during their trips, whereas *NR* cases mean that no re-routing is allowed. We use below the terms *vehicle* and *node* interchangeably.



(a) Low traffic demand (b) High traffic demand

Figure 7: Adjacency matrices for vehicular traces. Light-red (*dark-red*) color depicts low (*strong*) nodes' connectivity. R (*NR*) means all (*none*) vehicles can re-route their trips.

Fig. 7(a) shows that under a low traffic demand, the *NR* case obtains a set of nodes that are highly interconnected (there are few white spaces), while the *R* case represents a sparser nodes' connectivity. The reason is that in case R vehicles are allowed to choose alternative routes to avoid congested roads. Also, vehicles move faster, reducing connectivity opportunities. This way, we can notice how the re-routing capacity of vehicles affects negatively the vehicular network connectivity under low traffic demands.

In contrast, under high traffic demands, see Fig. 7(b), the R case shows a large amount of connected nodes with high intensity. The reason is that as the traffic demand increases, the probability of a vehicle to be in a traffic jam increases too. In such a case, vehicles would move slowly so that nodes increase their connectivity opportunities. However, for very high traffic demands traffic jams are unavoidable, which could also lead to a congested vehicular network with a high packet collision probability. In this case, re-routing capacity of vehicles can improve the performance of the vehicular network since traffic jams can be alleviated and thus the chance of packet collisions decreases.

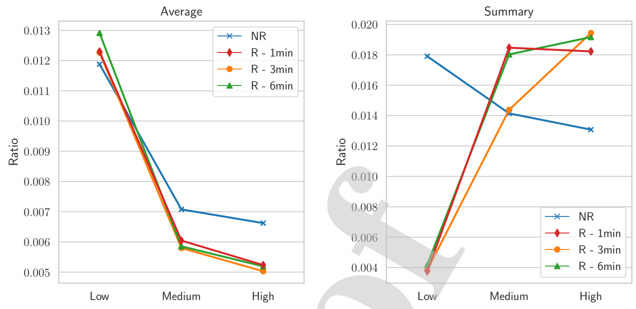
In the rest of this section, we analyze the nodes' connectivity in terms of well-known graph metrics to characterize vehicular networks. For this, we define two scenarios called *Average*, and *Summary*, as follows:

Average graph: We construct a graph G_t for every snapshot. Then, we average the metrics computed for each graph separately. We generate a network graph every $30s$, where the edges' weight is the strength of the signal. Recall that an *edge* of the graph links two *vertices* (vehicles) are in the transmission range of each other. This metric provides an idea of the behavior of the vehicular network at a specific moment.

Summary graph: We construct a single *summary* graph G considering all the samples during a period. In case two nodes were in the transmission range of each other (at least once during the simulation), we draw an edge between them. Weights of edges are obtained by adding the signal strength each time two nodes are in contact. This metric gives an idea of how often two nodes coincide during the studied period.

Edge percentage (EP): The EP is defined in the range [0, 1], where EP=0 means no connectivity between nodes and EP=1 means full connectivity between nodes. The EP is defined as $EP = \frac{E}{N \cdot (N-1)}$. Here, E is the number of edges and N the total number of vertices (nodes) of G. Fig. 8 shows the *average* and *summary graphs* detailed above. Regarding the *average* graph, all re-routing cases produce a similar behavior. The ED is low, just 1,3% of vehicles are (directly) connected under sparse scenario (Low traffic); furthermore, this index halves under high traffic condition. This effect is more notable when re-routing is allowed, since vehicles can modify their planned routes to avoid congested streets so they will find less vehicles in the new routes.

Regarding the *summary graph*, in case of low traffic demand by enabling re-routing capabilities (R cases) nodes' connectivity between nodes is highly affected (negatively) compared to the NR case (no re-routing is allowed). This is because during the studied period, vehicles select new routes with low congestion levels so that vehicles are sparsely distributed and consequently the EP decreases. In contrast, with high traffic demand the EP values increase (to a higher value than the NR case), meaning that it is possible to establish more connections



(a) (b)

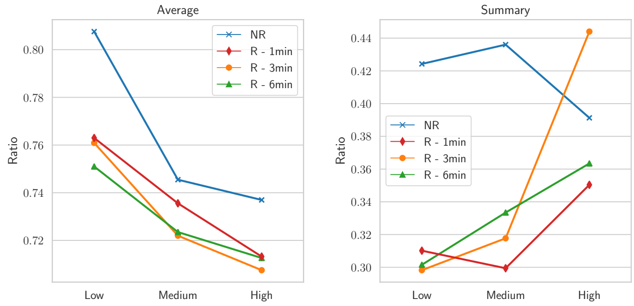
Figure 8: Edge percentage (EP) to measure the connectivity level between nodes (vehicles). (a) Average graph, (b) summary graph. Vehicles can re-route every R min using the RT traffic generation tool

.

between nodes compared to the NR case (no re-routing is allowed). The reason is that vehicles looking for a new route will try to look for a faster route, but also with a lot of traffic. The results of the *summary graph* correspond with observed behavior with the adjacency matrices in Fig. 7. That is, the re-routing capacity of vehicles negatively affects the vehicular network connectivity under low traffic demands.

Transitivity (T): The transitivity of a graph is defined as the ratio of *triangles* in the graph (i.e., $a \to b, b \to c, c \to a$), compared to the total number of *connected triples* (i.e., $a \to b, b \to c$). T indicates the proportion of nodes that have adjacent nodes interconnected. It is defined in the range [0, 1], where T=1 if the network contains all possible edges (full connectivity). In opportunistic networks, transitivity can be interpreted as the chances that two nodes that communicate directly ($a \to b$) can also communicate back by using only two additional hops ($b \to c, c \to a$). In Fig. 9, the higher associated transitivity takes place with NR under low and medium traffic. We can conclude that by enabling re-routing capabilities, less back-up triangle paths will be generated. This is due to the fact that dynamic route planning diminishes congestion levels throughout the road map. In this case, vehicles move faster so that it is less probable that back-up triangles can be generated.

Diameter (D): The diameter of a graph G is the length of the longest shortest path (obtained in terms of the number of hops) between any two graph vertices (nodes) i and j in G. This way, D reflects the largest number of vertices which must be traversed in order to travel from one vertex to another (paths which backtrack, detour or loop are not considered). Regarding the *average* scenario in Fig. 10, we can see that as the traffic demand increases, diameter values for R-cases (re-routing is allowed) also increase. In the context of vehicular communications, a long diameter value means that distant nodes could be reached using a suitable routing protocol. This result agrees to the one obtained for the edge percentage, see Fig. 8. A low EP value means that more hops are needed to reach distant nodes. Regarding the *summary* case, we can see that for low traffic density, as the re-routing interval decreases (i.e., re-routing events occurs more often), the diameter increases. Particularly, in the

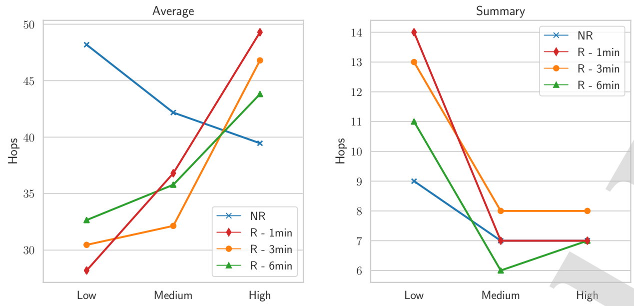


(a) (b)

Figure 9: Normalized transitivity (T), where T=1 if the network contains all possible edges (full nodes' connectivity). Vehicles can re-route every R min using the RT traffic generation tool

.

"R - 1 min" case, vehicles cover more areas of the map (vehicles are spread distributed), so that vehicles will require more hops to connect to distant vehicles. In case of medium and high traffic demands, no significant difference is produced for the three re-routing intervals considered.

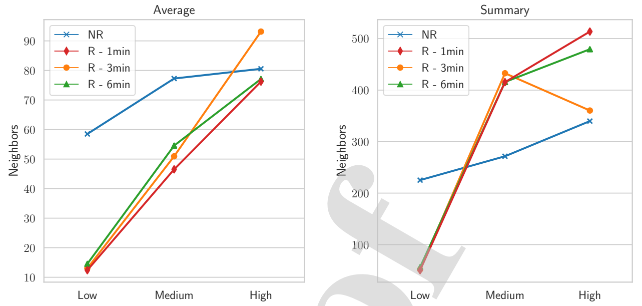


(a) (b)

Figure 10: Diameter (D), measured as the hopcount. Large D values mean vehicles will require more hops to connect to distant vehicles. Vehicles can re-route every R min using the RT traffic generation tool

Mean node's degree: The degree of a node i is defined as $deg(i) = |N(i)|$, where $N(i)$ is the number of neighbors of that node. In Fig. 11, as it is expected when the traffic demand increases the average nodes' degree increases as well (i.e., nodes have a higher number of neighbors). Focusing on the *average* case, we can see that while the NR case obtains higher node degree values when traffic demand is low and medium, with high traffic demand all cases behave similarly. Regarding the *summary* case, re-routing options produce higher values for medium and high traffic demands. It means that nodes will have more options to forward packets under high traffic demands for all cases. However, very high traffic demands can lead to congested roads (traffic jams) and also to congested vehicular network.

Betweenness centrality: It measures the number of shortest paths that go through a particular node. This metric shows if there are nodes that often become intermediate nodes between

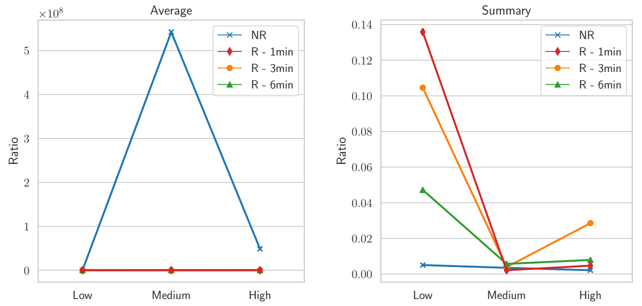


(a) (b)

Figure 11: Mean nodes' degree. It shows the average number of neighbors. Vehicles can re-route every R min using the RT traffic generation tool

.

other nodes in the network. We use the signal strength (i.e., edges' weight) to compute the shortest paths. In Fig. 12, a high value shows that a set of nodes arises repetitively in the shortest path creation. We can see that the NR option (no re-routing is allowed) reaches higher values for medium and high traffic demands. This is because vehicles mainly follow the same routes in their trips (computed in an empty road map) so that it produces similar intermediate nodes. On the other hand, among the "R cases" (re-routing is allowed), there is not a significant difference, since vehicles choose different routes based on traffic conditions. It is important to notice that the centrality score accounts for the disproportion of the metric (in this case betweenness) among nodes of the graph. In this regard, medium traffic load leads to the highest disproportion in the score of nodes. However, the average betweenness value of a node in high traffic is higher than in the medium traffic case.

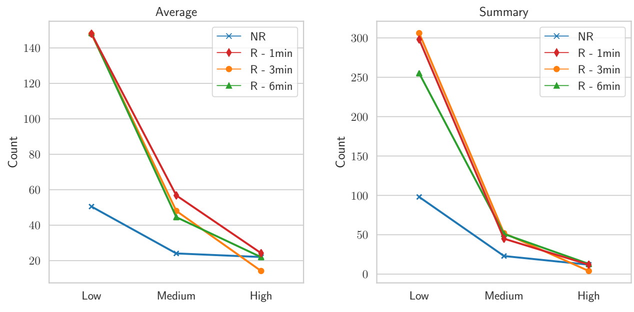


(a) (b)

Figure 12: Betweenness centrality. It highlights if there are nodes that often become intermediate nodes between other nodes in the network. Vehicles can re-route every R min using the RT traffic generation tool

.

Number of Communities: This metric shows the tendency of a graph with nodes densely connected among each other, but sparsely connected to other communities. This metric is useful to infer how clusters are formed when routing protocols based on cluster-approach work. To detect the communities, we use the algorithm proposed in [28] for every graph. In Fig. 13, we can see that a higher number of clusters are obtained for "R cases", especially when vehicles are more sparsely distributed. As it is expected, the number of clusters decreases as the traffic demand increases since more often vehicles encounter each other during the simulation.



(a) (b)

Figure 13: Number of communities. It shows the tendency to form clusters in the network. Vehicles can re-route every R min using the RT traffic generation tool

.

5.1. Discussion

In the context of vehicular networks, to evaluate new developments (e.g., routing protocols, ITS services) using simulation frameworks, the use of a realistic traffic mobility model is of utmost importance to obtain accurate results. In this regard, the output vehicular traces obtained with the diverse SUMO tools present fundamental differences, as it is detailed in Section 4. Furthermore, by enabling re-routing capabilities, vehicles dynamically modify their routes during the simulation altering the overall traffic mobility and the nodes' connectivity. Considering low traffic demands (off-peak hours), re-routing features highly impact on the nodes' connectivity due to the fact that selected alternative routes are those with less crowded roads. In case of high traffic demand (peak hours), re-routing features improve the vehicular network connectivity motivated by the fact that vehicles look for free routes, which produces an expansion of the network coverage (i.e., vehicles are widely distributed).

6. Proposed SUMO traffic generation tool (STG)

Even-though the SUMO framework facilitates the traffic generation with its build-in tools (e.g., OD2Trips, RandomTrips, DUArouter, DUAIterate, MARouter), the overall traffic demand generation may result in a complex and time-consuming process. To build route traces, SUMO tools require several input files, as it was detailed in Section 4.1. Moreover, in order to enable re-routing functionality, additional configurations are required. In this regard, to generate the traffic demand we have used our **SUMO traffic generation tool (STG)**, which is available in [29]. The STG tool facilitates the traffic demand generation based on SUMO tools. It allows researchers to quickly execute the different SUMO tools to select the appropriate traffic generation tool for a given scenario (e.g., urban, highway).

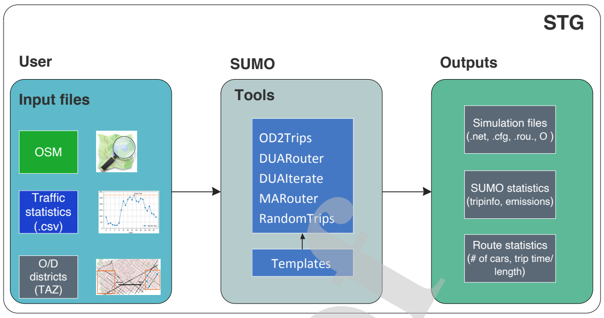


Figure 14: SUMO traffic demand generation (STG) tool workflow.

```
Usage: stg run [OPTIONS]

  STG SUMO Traffic generator. Required options: tool, -O, -D,

Options:
  -s, --sumo-bin PATH             SUMO bin directory.
  -osm PATH                       OpenStreetMap file (.osm)
  -T, --real-traffic PATH         Path to real traffic file with .csv format.
  -O, --O-district-name TEXT      Origin district name as in TAZ file.
  -D, --D-district-name TEXT      Destination district name as in TAZ file.
  -o, --outputs PATH              Output directory (route traces, statistics).

  -ma                             MARouter SUMO tool
  -dua                            DUARouter SUMO tool
  -duai                           DUAIterate SUMO tool
  -rt                             RandomTrips SUMO tool
  -od2                            OD2Trips SUMO tool
  -gui                            Graffical interface for SUMO simulations

  -p, --max-processes INTEGER     The maximum number of parallel simulations. [
                                  default available cpus are used ]
  -t, --sim-time INTEGER          Number of hours to simulate  (e.g., 24 hours)
                                  [default: 1]
  -n, --repetitions INTEGER       Number of repetitions.  [default: 1]
  --help                          Show this message and exit.
```

Figure 15: SUMO traffic demand generation (STG) tool command line interface.

For this, the STG tool automatically generates the required files to execute the different SUMO tools. Additionally, the STG tool uses real traffic statistics to create route traces as realistic as possible. Here, the user can configure the period in which the traffic is generated (e.g., full-day traffic).

The overall STG workflow is shown in Fig. 14. First, the users' domain includes the input files: (i) the sumo network file, (ii) the traffic assignment zone (TAZ) file including the origin and the destination districts. Optionally, the user can select actual traffic statistics to generate a realistic and validated traffic demand. This file contains the traffic intensity between the defined origin-destination districts. Here, the traffic intensity is defined by the number of vehicles per simulation hour. Then, the SUMO domain includes the traffic generation tools. The STG tool extends required files based on pre-configured templates. Finally, the STG tool outputs include simulations (console or GUI) and SUMO statistics. The STG tool is provided with a CLI interface as it is shown in Fig. 15.

7. Conclusions and future work

In this work, we have studied the impact on traffic mobility and nodes' connectivity when using the different SUMO tools (OD2, MAR, DUAR, DUAI, RT). First, we have analyzed the vehicle traces generated by each tool. Results show that tools present fundamental differences regarding route characteristics, emissions and hardware requirements. Besides, we have analyzed the nodes' connectivity by using well-known graph metrics for network analysis. Results show that vehicular network connectivity is highly impacted by enabling re-routing capabilities, especially in low traffic demands (off-peak hours). Simultaneously, it improves the connectivity of the vehicular network under high traffic demands (peak hours). Concluding, the selected traffic generation tool with re-routing capabilities would significantly modify the simulation results obtained when we assess the performance of services, such as traffic reporting and accident warnings. In addition, we can see that enabling re-routing capabilities situations like network deadlocks can be circumvented. Therefore, considering re-routing is essential to produce meaningful evaluation results when assessing ITS services.

Finally, traffic demand generation may result in a complex and time-consuming process because of required configuration files. Therefore, in the last section of this work, we propose a simple SUMO traffic generation tool called STG that facilitates the overall process and aims to help researchers in the traffic demand generation process.

In future work, we will assess a more extensive map area, including accidents during the simulation, to analyze the impact of route updates to avoid the area around the accident.

References

[1] C. Sommer, R. German, F. Dressler, Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis, IEEE Transactions on Mobile Computing (TMC) 10 (1) (2011) 3–15. doi:10.1109/TMC.2010.133.

[2] A. Hegde, A. Festag, Artery-C: An OMNeT++ based discrete event simulation framework for cellular V2X, in: Proceedings of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 47–51. doi:10.1145/3416010.3423240.
URL https://doi.org/10.1145/3416010.3423240

[3] SUMO Wiki, Demand/Introduction to demand modelling in SUMO - SUMO Doc. (2018).
URL https://sumo.dlr.de/docs/Demand/Introduction_to_demand_modelling_in_SUMO.html

[4] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, E. Wießner, Microscopic traffic simulation using sumo, in: The 21st IEEE International Conference on Intelligent Transportation Systems, IEEE, 2018, pp. 2575–2582.
URL https://elib.dlr.de/127994/

[5] Ajuntament de Barcelona, Estudi de mobilitat de la Travessera de Gràcia (2021).
URL https://bcnroc.ajuntament.barcelona.cat

[6] Ajuntament de Barcelona, Anuario Estadístico de la Ciudad de Barcelona (2019).
URL https://www.bcn.cat/estadistica/castella/dades/anuari/cap15/index.htm

[7] European Union, EUR-Lex - l28186 - EN - EUR-Lex (2014).
URL https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=LEGISSUM{%}3Al28186http://eur-lex.europa.eu/legal-content/DE/TXT/?uri=URISERV:l28186

[8] J. Benavides, M. Guevara, M. G. Snyder, D. Rodríguez-Rey, A. Soret, C. P. García-Pando, O. Jorba, On the impact of excess diesel NOx emissions upon NO2 pollution in a compact city, Environmental Research Letters 16 (2) (2021) 24024. doi:10.1088/1748-9326/abd5dd.
URL https://doi.org/10.1088/1748-9326/abd5dd

[9] L. Codeca, R. Frank, T. Engel, Luxembourg sumo traffic (LuST) scenario: 24 hours of mobility for vehicular networking research, in: 2015 IEEE Vehicular Networking Conference (VNC), IEEE, 2015, pp. 1–8.

[10] M. Berrazouane, K. Tong, S. Solmaz, M. Kiers, J. Erhart, Analysis and initial observations on varying penetration rates of automated vehicles in mixed traffic flow utilizing sumo, in: 8th Int. Conf. on Connected Vehicles and Expo, 2019.

[11] L. Codecá, J. Härri, Towards multimodal mobility simulation of c-its: The monaco sumo traffic scenario, in: 2017 IEEE Vehicular Networking Conference (VNC), 2017, pp. 97–100.

[12] J. L. Zambrano-Martinez, C. T. Calafate, D. Soler, J.-C. Cano, Towards realistic urban traffic experiments using dfrouter: Heuristic, validation and extensions, Sensors 17 (12). doi:10.3390/s17122921.
URL https://www.mdpi.com/1424-8220/17/12/2921

[13] Y. Huang, M. Chen, Z. Cai, X. Guan, T. Ohtsuki, Y. Zhang, Graph theory based capacity analysis for vehicular ad hoc networks, in: 2015 IEEE Global Communications Conference (GLOBECOM), 2015, pp. 1–5.

[14] B. Amina, E. Mohamed, Performance evaluation of VANETs routing protocols using sumo and NS3, in: 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), 2018, pp. 525–530.

[15] J. Nzouonta, N. Rajgure, G. Wang, C. Borcea, VANET routing on city roads using real-time vehicular traffic information, IEEE Transactions on Vehicular Technology 58 (7) (2009) 3609–3626.

[16] C. Steve, OpenStreetMap (2020).
URL http://www.openstreetmap.org

[17] SUMO Wiki, ACTIVITYGEN - SUMO Doc. (2018).
URL https://sumo.dlr.de/docs/ACTIVITYGEN.html

[18] SUMO Wiki, NETCONVERT - SUMO Doc. (2018).
URL https://sumo.dlr.de/docs/NETCONVERT.html

[19] C. Steve, OpenStreetMap (2020).
URL https://sumo.dlr.de/docs/index.html#application_manuals

[20] SUMO Wiki, DUAROUTER - SUMO Doc. (2018).
URL https://sumo.dlr.de/docs/DUAROUTER.html

[21] SUMO Wiki, Demand/Automatic Routing - SUMO Doc. (2018).
URL https://sumo.dlr.de/docs/Demand/Automatic_Routing.html

[22] SUMO Wiki, Developer/How To/Device - SUMO Doc. (2018).
URL https://sumo.dlr.de/docs/Developer/How_To/Device.html

[23] Dirección General de Tráfico DGT, Estadísticas e indicadores (2019).
URL https://www.dgt.es/es/seguridad-vial/estadisticas-e-indicadores/

[24] D. Krajzewicz, M. Behrisch, P. Wagner, R. Luz, M. Krumnow, Second generation of pollutant emission models for SUMO, in: M. Behrisch, M. Weber (Eds.), Lecture Notes in Control and Information Sciences, Vol. 13, Springer International Publishing, Cham, 2015, pp. 203–221. doi:10.1007/978-3-319-15024-6_12.

[25] T. Johnson, A. Joshi, Review of vehicle engine efficiency and emissions, SAE International Journal of Engines 11 (6) (2018) 1307–1330.
URL https://www.jstor.org/stable/26649163

[26] M. Newman, Networks, Oxford university press, 2018.

[27] C. Sommer, D. Eckhoff, R. German, F. Dressler, A computationally inexpensive empirical model of IEEE 802.11p radio shadowing in urban environments, 2011 8th Int. Conf. on Wireless On-Demand Network Systems and Services (2011) 84–90doi:10.1109/WONS.2011.5720204.

[28] A. Clauset, M. E. J. Newman, C. Moore, Finding community structure in very large networks, Phys. Rev. E 70 (2004) 066111. doi:10.1103/PhysRevE.70.066111.
URL https://link.aps.org/doi/10.1103/PhysRevE.70.066111

[29] P. Barbecho, SUMO traffic generator (STG) (2021).
URL https://github.com/Pbarbecho/STG.git

Pablo Barbecho and Luis Felipe Urquiza: Conceptualization, Methodology, Software development, Writing, Original draft preparation, Visualization, Investigation

Mónica Aguilar Igartua: Conceptualization, Validation, Writing-Reviewing and Editing.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: